JavaScript Functions

Function is a collection of statements to perform a task.

```
function FuncName ()
        {
                document.write("New Function");
        }
```

JavaScript Functions

Calling a function:

FuncName ();

```
<html>
<head>
<title> Functions: user defined </title>
<script type="text/javascript">
function disp_alert()
{
alert("I am an alert box!!")
}
</script>
</head>
```

```html
<body>
<form>
<input type="button" onclick="disp_alert()" value="Display alert box">
</form>
</body>
</html>
```

Function Parameters

```
function getArea (width, height)

{

        return width*height;

}



Calling a function – getArea(3,5);
```

Function Parameters

```
function getArea (width, height)
{
        var area=width*height;
        return area;
}


var wallOne=getArea(3,5);
var wallTwo=getArea(8,5);
```

Function Parameters

```html
<html>
<head>
<title>welcome</title>
<script language-"JavaScript">
function getArea (width, height) {
        var area=width*height;
        return area;
}
```

Function Parameters

```
var wallOne=getArea(3,5);

var wallTwo=getArea(8,5);

document.write(wallOne + "," + wallTwo);


</script>

</head>

<body>

</body>

</html>
```

Functions

User defined
Predefined alert prompt

parsInt   converts variable into an integer
parseFloat   converts variable into a number
Math.sqrt   square root
Math.floor   rounding to the lower integer
Math.round   rounding

School of Design

Functions

- document.write()

- close()

- getElementById()

- getElementsByName()

- getElementsByTagName()

- open()

- Writes HTML expressions or JavaScript code to a document

- Closes an output stream opened with the document.open() method, and displays the collected data

- Returns a reference to the first object with the specified id

- Returns a collection of objects with the specified name

- Returns a collection of objects with the specified tagname

- Opens a stream to collect the output from any document.write() method

JavaScript example: hello.html

```html
<html>
<head>
<title>hello</title>
<script>
document.write("Hello class!");
</script>
</head>
<body>
</body>
</html>
```

DOM

Object-oriented (OO):

Instead of writing procedural programs, write class libraries to encapsulate behaviors

DOM - The Document Object Model

a cross-platform and language-independent API that binds JavaScript and other scripting languages together with HTML and other markup languages

DOM

is not a collection of dumb elements but a hierarchy of types
Styles are properties of objects

Complete OO code with error handling, instance methods, static methods and type hierarchies

Versatile use of functions

A large number of object-oriented libraries
Used to create User Interfaces

```
<html>
<head>
<title>bgcolor_change</title>
<script language="JavaScript">
function newbg(paramColor)
{
document.bgColor=paramColor;
}
</script>
</head>
```

Functions : bgcolor.html

```html
<body textcolor="black" link="black" alink="black">
<center>
<a href="#" onmousedown="newbg('olive');"> olive</a><br />
<a href="#" onmousedown="newbg('blue');"> blue</a><br />
<a href="#" onmousedown="newbg('Beige');"> beige</a><br />
<a href="#" onmousedown="newbg('yellow');">yellow</a><br />
</center>
</body>
</html>
```

Exercise

Add 3 background colors of your choice to this example